Third International Symposium on Computational Mechanics (ISCM III)
Second Symposium on Computational Structural Engineering (CSE II)
December 5-7, 2011, Taipei, TAIWAN

An Efficient Tetrahedral Mesh Generation Approach for Boundary Face Method

Cheng Huang *, Jianming Zhang †

* State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha 410082, China (huangcheng324@163.com)
† State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha 410082, China (zhangjianm@gmail.com)

This paper presents a new efficient Delaunay tetrahedral mesh generation method coupled with the advancing front method. The algorithm combines the advantages of the high quality point placement of the advancing front method and the high efficiency and convergence guaranty of the Delaunay algorithm. The method that combined the Delaunay and advancing front method has been described first by Rebay [1] and Miller [2]. And Pascal J. Frey et al. extend this method to three dimensions [3]. In our work, we use a new efficient front identification and point generation procedure to further improve the efficiency.

First step of the mesh algorithm is the boundary mesh generation. At first, we construct a bounding box enclosing the domain. The box is defined by eight vertices and divided into five tetrahedrons. All the boundary points are inserted using the Delaunay kernel. Then, the boundary is regenerated. The second step is to generate the three dimension tetrahedron mesh. The triangle of the boundary triangle discretization is defined as the initial front. The initial front is identified to get the active front. Similar to the conventional advancing front method [4], we define the triangle generated in the last step as the front and the front which can use to generate new points as the active front. The internal points are iteratively created by the active front and inserted into the previous mesh by Delaunay kernel. Repeat the creation loop as long as the active front is not empty. After the mesh generation, the Laplacian smoothing and topological optimization process are used to improve the quality of the mesh.

In our approach, a set of new triangle is created after the insertion of points at stage i. Then, the front is identified in order to obtain the active front. At stage i+1, the active front is used to create new field points and generate new tetrahedral elements. Let F_i be a triangle of the set of new triangle created at stage i. If F_i belongs to two new tetrahedrons created at stage i, it is added to the set of front. Circumcircle center R of the front is calculated and mesh size H of the front's center point is extracted from the control space. If H > R, the front is added into the set of active front. If H < R, the new point can not generate in the two sides of the front, and the front is eliminated.

After the front identification, we get a set of active front. Assume $\triangle P_1P_2P_3$ is one triangle of the active front. In three dimensions, we always want get regular tetrahedron. Assume h is the requested size. So the length of the tetrahedron's edge equals to h, as in figure 1. We have to calculate the location of point N. Let point C be the circumcircle center of $\triangle P_1P_2P_3$. We can prove that line NC is perpendicular to the plane $P_1P_2P_3$. Then, we just have to calculate the location of point C to get the coordinate of point N. Let us assume now that we know the coordinates of points $P_1(X_0, Y_0, Z_0)$, $P_2(X_1, Y_1, Z_1)$, $P_3(X_2, Y_2, Z_2)$. We can calculate the coordinate of point C by solve the following equations. $\mathbf{n}(n_1, n_2, n_3)$ is the normal vector of the plane $P_1P_2P_3$.

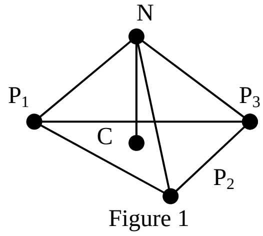

Figure 1

$$AX = B \qquad X = \begin{bmatrix} x_c & y_c & z_c \end{bmatrix}^T$$

$$A=\begin{bmatrix} n_x & n_y & n_z \\ x_1-x_0 & y_1-y_0 & z_1-z_0 \\ x_2-x_0 & y_2-y_0 & z_2-z_0 \end{bmatrix} \qquad B=\frac{1}{2}\begin{bmatrix} 2(n_x x_0+n_y y_0+n_z z_0) \\ x_1^{\,2}+y_1^{\,2}+z_1^{\,2}-x_0^{\,2}-y_0^{\,2}-z_0^{\,2} \\ x_2^{\,2}+y_2^{\,2}+z_2^{\,2}-x_0^{\,2}-y_0^{\,2}-z_0^{\,2} \end{bmatrix}$$

Then we can calculate the circumcircle's radius r of the $P_1P_2P_3$.
The length of the new point to the plane $P_1P_2P_3$ is $d= (h^2 – r^2)^{0.5}$.
The location of the new point N can calculate by $(X_n, Y_n, Z_n)=(X_c, Y_c, Z_c) \pm dn$.
New points need to be filtrated. As every new point is created apart from the others, a filtration process can remove appoint which violates the size criterion when compared with a previously selected point.
The new point is accepted, if it satisfies the following criterion:
1. Let L(NP) be the length between new point N and arbitrary point P. N is filtered if the length L(NP)<0.8h.
2. Let T be the tetrahedron which links to the current triangle in the same side of the new point N. Let L be the average length of three edge of the tetrahedron different from the edge of the current triangle. N is filtered if L > h.
We use the background mesh for specifying mesh size distributions over the problem domains. The background mesh is defined by the boundary mesh of the domain. To obtain the mesh size of an arbitrary point within the domain, it is only needed to define the size at every vertex of background mesh. The mesh size at some arbitrary point within the domain is calculated by linear interpolation of the nodal values of the element in which the point is contained.

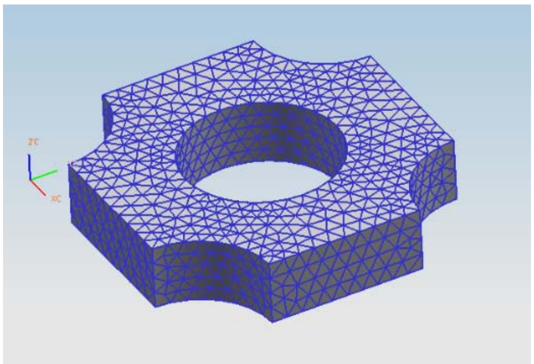

Figure 2

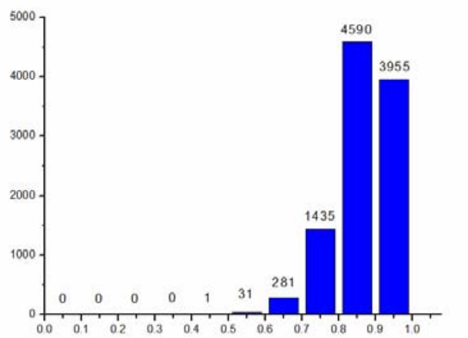

Figure 3

An example is selected to demonstrate the robustness and efficiency of the proposed method. The example is drawn in Figure 2. The mesh consists of 2300 surface elements, 10293 volume elements, and 2362 mesh nodes. Figure 3 shows the quality classes of the volume elements. The total CPU time need for the generation of the mesh was 2 seconds.

References

[1] S. Rebay. Efficient unstructured mesh generation by means of Delaunay triangulation and Bowyer/Watson algorithm. 3rd Int. Conf. on Numerical Grid Generation in Comp. Fluid Dyn., Barcelona, Spain, June 1991.

[2] J.D. Muller, P.L. Roe and H. Deconinck. A frontal approach for node generation in Delaunay triangulations, Unstructured grid methods for advection dominated flows, VKI lecture notes, pp. 9-1-9-7, AGARD Publication R-787, March 1992.

[3] P. J. Frey, H. Borouchaki, P.L. George. 3D delaunay mesh generation coupled with an advancing front approach. Comput Meth Appl Mech Engrg. 1998. 157 :115-131.

[4] Jan Frykestiong. Advancing front mesh generation technique with application to the finite element method. Department of Structural Mechanics of Chalmers University of Technology, 1994, 10.